**Blogovací redakční systém**

Tento systém jsem si vybral, protože ve své práci mám na starosti weby, které jsou hlavně zaměřeny na blogy. Tyto weby běží na systému Wordpress. Je to komplexní systém a díky tomu může mít pomalé načítání a je zbytečně komplexní pro malé weby.

Jsem nejvíce seznámený s programovacím jazykem PHP, proto jsem si vybral systém, který je na PHP. Nejdříve jsem tento projekt zpracovat na frameworku Laravel ale po přečtení několika částí dokumentace jsem zjistil, že je to velice komplexní systém. Díky tomu jsem využil Framework Yii2, který mi byl doporučený kolegou. YII2 je rozděleno pro základní použití a pokročilé použití.

V tomto projektu je využita základní verze, která má spojený front-end a back-end. První co jsem zpracoval je jejich rozdělení. Je pouze částečné, protože mají stejnou konfiguraci. Stejná konfigurace může vyvolávat problémy s routováním, proto bude vhodné před dokončením vylepšit na pokročilou verzi Yii2.

Yii pracuje s databázovými tabulky jako s modelem. Každá tabulka má vlastní model, přes který se přistupuje k databázi. Tomuto modelu nemusíme nastavovat vůbec nic, pouze to, že rozšiřuje třídu ActiveRecord.

Po rozdělení jsem zpracovával administraci. Prvním krokem bylo připojit databázi, tabulky a vytváření uživatelů. Dlouho jsem zjišťoval, jaké šifrování hesla je nejlepší a nakonec jsem použil šifrovací funkci php se Salt proměnnou. Každé heslo má vygenerován svůj vlastní Salt.

Hlavní možností tohoto systému je správa příspěvků, kategorií a uživatelů. K příspěvkům a kategoriím je využita druhá databázová tabulka, do které se ukládají další informace (meta informace). Před tyto informace jsou propojeny příspěvky s kategoriemi.

Když byly základní funkce administrace hotové, tak bylo třeba nastylovat back-end prostředí. Ke stylování jsem využil Less css, které umožňuje příjemnější zápis css informací a také definici proměnných.

Následně se vytvářelo front-end prostředí, které umožňuje ukládání pouze pro komentáře. Pro všechno ostatní je to pouze zobrazování a čtení dat z databáze. Většina url adress je zpracována přes url managera a htaccess, které umožňují poslat GET informace, aniž by se zobrazovaly v url. Díky tomu je takové zpracování bezpečnější. Po vytvoření bylo třeba toto prostředí nastylovat. Znovu bylo využito Less css.

Po těchto krocích je hotová základní funkčnost a bude se přecházet k opravám a dalším funkcím. Funkce, které mi daly nejvíce zabrat, byly: šifrování hesel, nastavení routování v url manageru a správné zabezpečování formulářů.

V tomto stavu v jakém je se dá aplikace použít, ale není kompletně vyřešené nahrávání obrázků.

Plánované rozšíření:

* API soc. sítí – sdílení, share a možnost Facebook komentářů.
* Kompletní správa mediálních souborů – Problém je, že lidé nenahrávají obrázky v přesném rozlišení. Proto mám v plánu zpracovat kompletní správu, ve které se nahraje obrázek v maximální velikosti, a uživatel si bude moci vybrat výřezy a ty poté uložit jako menší obrázky. Této funkce se bude využívat například u thubnailů pro příspěvky a obrázků pro slider.
* Přesun všech vlastních nastavení z config souborů do databáze (povolování komentářů, nastavení slideru, nastavení menu, nastavení widgetů, atd.)

Na projektu jsem strávil přes 35 hodin čistého času a celková cena by se pohybovala kolem 5 000,- Kč v tomto zpracování. Za tuto cenu by ovšem nebyl prodáván, byla by stanovena cena za využití, která by se pohybovala někde kolem 500,- Kč.